IN THE CLAIMS

This listing of claims will replace all prior versions and listings of claims in the Application:

1. (Original) A method of generating a global history vector comprising the steps of:

determining if a selected group of instructions contains a branch instruction;

maintaining a current global history vector in a shift register when the selected group does not contain a branch instruction;

shifting a first value into the shift register to generate a second vector if the selected group contains a branch instruction and the branch instruction is predicted as a branch taken; and

shifting a second value into the shift register to generate a second vector when the selected group contains a branch instruction and the selected group does not include a branch instruction predicted as a branch taken.

2. (Original) The method of Claim 1 and further comprising the step of storing the generated value in an entry in a branch instruction queue associated with the selected group of instructions.

3. (Original) The method of Claim 2 and further comprising the step of correcting the generated vector upon a misprediction comprising the substeps of:

retrieving a selected number of bits of the vector stored from the branch instruction queue into the shift register; and

shifting an updated history bit into the shift register.

4. (Original) The method of Claim 1 wherein the first value comprises a logic 1 and the second value is a logic 0.

5. (Original) The method of Claim 1 wherein the selected group of instructions

comprises eight instructions.

6. (Original) A method of performing branch predictions comprising the steps of:

indexing a branch history table using a first global history vector associated with a first fetch group of instructions during a first fetch cycle to retrieve a first prediction value;

generating a second global history vector associated with a second fetch group of instructions comprising the substeps of:

retaining the first vector when the first fetch group does not contain at least one branch instruction;

appending a bit of a first value to the first vector when the first fetch group has at least one branch instruction predicted to be a branch taken; and

appending a bit of a second value to the first vector when the first group contains at least one branch instruction and contains no branch instructions predicted to be a branch taken; and

indexing the branch history table using the second global history vector during a second fetch cycle to retrieve a second branch prediction value.

7. (Currently amended) The method of Claim 6 and further comprising the step of storing the first and second vectors in an entry of a branch history queue associated ~~with the~~ with the first fetch group.

8. (Original) The method of Claim 7 and further comprising the steps of:

detecting a branch misprediction based on the first prediction value;

retrieving the first and second vectors from the branch history queue;

indexing the branch history table using the first vector to correct the first prediction value; and

appending a corrected bit to the second vector to generate a corrected branch history vector.

9. (Original) The method of Claim 7 wherein said first fetch cycle precedes the second fetch cycle by three fetch cycles.

10. (Original) The method of Claim 7 wherein said steps of indexing comprises the step of gating the vector with selected bits of a current instruction address.

11. (Original) The method of Claim 10 wherein said steps of gating comprise the steps of performing XOR operations.

12. (Currently amended) The method of Claim 8 wherein said substeps of appending comprise the substeps of shifting a bit into a shift register storing the ~~first~~ second vector.

13. (Original) Branch processing circuitry comprising:

a shift register for storing a global history vector;

control circuitry for selectively updating a first global history vector stored in said shift register operable to:

determine if a selected group of instructions contains a branch instruction;

maintain said first global history vector in said shift register when the selected group does not contain a branch instruction;

shift a first value into the shift register to generate a second vector if the selected group contains a branch instruction and the branch instruction is predicted as a branch taken; and

shifting a second value into the shift register to generate a second vector when the selected group contains a branch instruction and does not contain a branch instruction predicted as a branch taken.

14. (Original) The branch processing circuitry of Claim 13 and further comprising a branch history table and circuitry for generating an index to an entry in said branch

history table using selected bits from a current address and selected bits of said first vector to retrieve a prediction value stored therein.

15. (Original) The branch processing circuitry of Claim 14 and further comprising circuitry for updating said second vector when said prediction value results in a misprediction comprising:

a queue for storing said first and said second vectors;

circuitry for accessing said vectors from said queue;

circuitry for indexing said branch history table with said first vector and updating a corresponding entry with a corrected prediction value; and

circuitry for updating a vector in said shift register with said second vector; and

circuitry for shifting the corrected prediction value into said shift register.

16. (Original) The branch processing circuitry of Claim 13 wherein said branch processing circuitry forms a portion of a single-chip microprocessor.

17. (Original) A processing system comprising:

a microprocessor comprising:

a branch history table for storing branch prediction values;

a global history shift register for storing a global branch history vector;

logic for generating an index to said branch history table and accessing prediction values stored therein using selected bits of a said branch history vector stored in said shift register; and

control circuitry for updating a said global branch history vector stored in said shift register and operable to:

retain a current vector stored in said shift register when a selected fetch group does not contain at least one branch instruction;

shift a bit of a first value into said shift register to generate an updated vector when the selected fetch group has at least one branch

instruction predicted to be a branch taken; and

shift a bit of a second value into said shift register when said selected fetch group contains at least one branch instruction and contains no branch instructions predicted to be a branch taken.

18. (Original) The processing system of Claim 17 wherein said microprocessor further comprises :

a branch instruction queue having a plurality of entries each associated with a said fetch group for storing at least first and second corresponding global history vectors;

circuitry for detecting a misprediction associated with a said prediction value retrieved from said branch history table and corresponding to said first global history vector in said branch instruction queue;

circuitry for retrieving said first vector from said branch instruction queue and accessing a corresponding entry in said branch history table to correct said prediction value stored therein; and

circuitry for retrieving and modifying said second vector to generate a corrected vector in said shift register.

19. (Original) The processing system of Claim 17 wherein said processing system further includes a system memory coupled to said microprocessor by a bus.

20. (Original) The processing system of Claim 17 wherein a said fetch group comprises eight instructions.